Dynamic Communication and Computation Resource Allocation Algorithm for End-to-End Slicing in Mobile Networks

Hong Xu[1], Zhou Tong[2,3], Hong Shen[1], and Tiankui Zhang[2](✉)

[1] China Telecom Co., Ltd., BeiJing Branch, Beijing 100010, China
[2] Beijing University of Posts and Telecommunications, Beijing 100876, China
[3] China Mobile Research Institute, Beijing 100053, China

Abstract. In the mobile network, to support business diversity and meet the differentiated needs of vertical industries, network slicing technology has emerged. Network slicing is required in both the core network and the radio access network, that is, to achieve end-to-end network slicing. Most of the current network slicing service requirements are dynamic, in end-to-end network slicing, a deep Q network (DQN)-based two-stage joint allocation algorithm for communication and computation resource is proposed to solve the problem of dynamic changes of network slicing data queues, radio channel status, and physical network topology. The dynamic resource allocation model of end-to-end slicing is constructed, and the dynamic joint allocation of communication and computation resource is carried out to maximize the overall utility of the network on a long-term scale. The dynamic migration of virtualization network function (VNF) and the flexible allocation of virtual network resources are realized according to the service state and quality of service (QoS) requirements of virtual network users. The simulation results show that the proposed algorithm can optimize the overall utility of the network on a long-term scale, improve the long-term average revenue, and reduce the average cost of the system.

Keywords: Network slicing · Resource allocation · Dynamic

1 Introduction

In the mobile network, the service types of different users have different demands on the network. The introduction of mobile network slicing is to meet the diverse needs of vertical industry. The mobile network allows heterogeneous services to coexist in the same network architecture through network slicing. End-to-end network slicing includes core network (CN) slicing and radio access network (RAN) slicing, spanning the infrastructure of radio access network and core

X. Wang et al. (Eds.): AICON 2021, LNICST 396, pp. 251–267, 2021.
https://doi.org/10.1007/978-3-030-90196-7_22

network [1]. The process of slicing the network involves the allocation of multiple resources such as communication resource, computation resource, and cache resource on the CN side and the RAN side. Network slicing can be considered as a virtualized private network in the network. Through technologies such as software defined network (SDN) and network functions virtualization (NFV) [2], network functions are customized and tailored according to the needs of business scenarios, and network resources are reasonably allocated [3].

The infrastructure of end-to-end network slicing includes both CN and RAN. Therefore, it is an important research direction to consider the resources on the CN side and the RAN side when allocating resources for network slicing. In [4], To extend the coverage of NFV technology from the CN side to the field of radio access network, the radio virtualization network function (VNF) placement problem in the RAN was formalized as an integer linear programming problem, and a heuristic algorithm for VNF placement named radio network mapping was proposed to allow mobile virtual network operators to use customized resource allocation solutions to implement VNF placement on the RAN side. [5] proposed a communication and computation resource joint allocation algorithm in end-to-end network slicing for ultra reliable low-latency communication (URLLC), which can effectively reduce the end-to-end latency of network slicing, and guarantee the reliability requirements of network slicing. [6] proposed a radio resource allocation algorithm for Service Level Agreement (SLA) contract rate maximization, which can achieve a better SLA contract rate on the premise of ensuring isolation between slices, additionally increase the number of service users.

At present, most of the network slicing business requirements are dynamic, therefore the resource allocation process in network slicing will be affected by the randomness and time-varying nature of the actual environment [7]. However, most of the work still stays in the fixed environment to optimize the instantaneous network performance index. [8] set that the wireless channel is fixed. On the premise that the parameters of the radio channel are fixed, they proposed a cell planning scheme to maximize the resource utilization of the radio communication network. The proposed scheme optimized the resource allocation between network slices. From the perspective of information-centric networks and services, [9] studied the service function chain and optimization of IVCN on both the data plane and the control plane. In the fog-enabled heterogeneous RAN, they proposed a heuristic method ivcn-rano based on the ant colony optimization algorithm to solve the NP-hard problem, aiming to optimize the mapping of VNF and virtual content placement.

Dynamic slicing business scenarios often have a certain life cycle. In the existing researches, there are few research on dynamic resource allocation of network slicing that optimizes average performance indexes on a long-term scale for dynamic scenarios, and most of the existing researches on dynamic resource allocation of network slicing lack consideration of the overall state of the network, but consider the partial state of the network to optimize a single performance index. [1] proposed an upper-level priority algorithm with delay-limiting over-supply prevention to adjust capacity and traffic allocation to minimize the "over-supply ratio" while still meeting tenants' delay constraints and service level agreement. [10] designed a learning-based dynamic network slicing adjustment strategy, which can significantly reduce the overall expansion cost of network slicing while ensuring the quality of service. [11] proposed a dynamic communication resource sharing scheme for single-layer homogeneous C-RANs with multi-tenants, which considers the priority of tenants and maximizes the network utility. In the dynamic environment, the feasibility of solving optimization problems commonly used in static scenes is reduced, and new methods need to be explored.

In summary, in recent years, most of the work on slicing resource allocation only considers CN slicing or RAN slicing, and few studies consider the resource allocation of the entire end-to-end network slicing. And the process of network slicing deployment involves the joint allocation of multiple resources, so it is necessary to consider the joint allocation of communication and computation resource of end-to-end network slicing. In addition, the current work has less research on the dynamic allocation of network slicing resources. In response to the above problems, this paper takes into account the life cycle management of network slicing, studies the dynamic joint allocation of communication and computation resource based on the end-to-end network slicing architecture, aiming at the dynamic changes of network slicing data queues, radio channel status, and physical network topology. An end-to-end sliced dynamic resource allocation model is constructed, and a two-stage communication and computation resource joint allocation algorithm based on deep Q network (DQN) is proposed to maximize the overall utility of the network on a long time scale, to optimize the overall utility of the network, improve the long-term average revenue, and reduce the average cost of the system.

2 System Model

This paper proposes an end-to-end network slicing dynamic resource allocation scenario, as shown in Fig. 1. Based on the mobile network end-to-end network slicing scenario with dynamic service requests, for dynamic network slicing services with a certain life cycle, considering the dynamic joint allocation of communication and computation resource in the end-to-end network slicing, this paper studies the use of online resource management technology to maximize the overall network utility on a long-term scale. On the CN side, the underlying server of the infrastructure layer provides multiple types of network resources including computation resources and bandwidth resources. The virtualization layer virtualizes the network resources and provides various types of virtual network resources on the cloud server. According to the service requirements of network slicing, the CN side realizes the flexible allocation of physical node resources and link resources in discrete time slots according to the current system status and formulates the dynamic migration strategy for VNF, in which the computation resource is node CPU resource and the communication resource are link bandwidth resource between nodes. On the RAN side, the appropriate number of physical resource blocks (PRB) and computation resource are allocated to each network slice in discrete time slots according to the current system status. In the access network, the computation resource is related to the data processing capacity of each remote radio unit (RRU), and the communication resources are related to the PRB in each RRU.

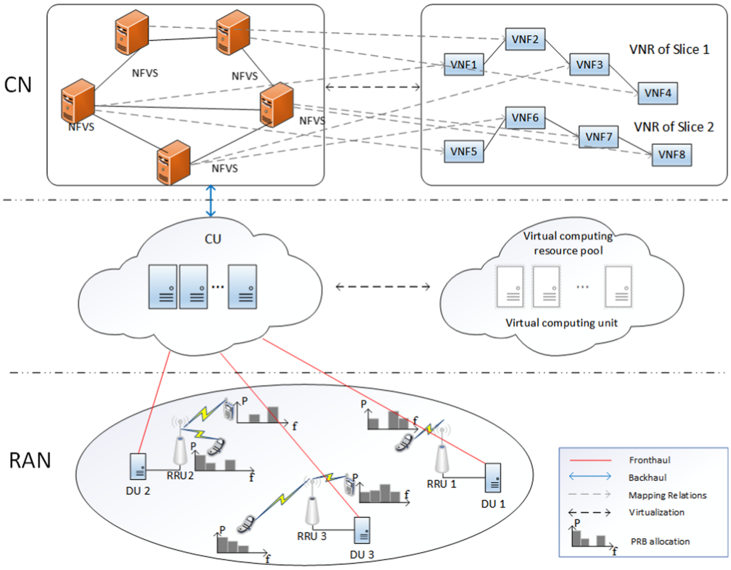

Fig. 1. End-to-end network slicing dynamic resource allocation scenario.

In this model, graph theory is used to describe the physical network and virtual network in the core network. The physical network can be expressed as an undirected weighted graph $G^P = \left(N^P, L^P, \eta_n^P, \eta_l^P\right)$, where N^P and L^P represent the set of physical nodes and the set of physical links between physical nodes respectively. In the core network, the set of general servers in the network is regarded as the set of physical nodes N^P, $N^P = \{1, 2, \cdots, N\}$. η_n^P represents the attribute set of physical nodes, such as CPU, memory, disk, etc. η_l^P represents the underlying link bandwidth. In the actual system, the specific resource requirements are usually related to the amount of data that the VNF needs to process.

Combine logical functions and generate virtual network topology according to service request. The virtual network request can be expressed as an undirected weighted graph $G^V = \left(N^V, L^V, \eta_n^V, \eta_l^V\right)$, where N^V represents the set of VNFs, that is, the set of virtual nodes; L^V represents the set of virtual links, η_n^V represents the attribute set of VNFs. The process of mapping VNF to the underlying network is expressed as $M : \left\{N^V \to N^P, L^V \to L^P\right\}$. The set of slices is $I = \{1, 2, \cdots, i\}$. The VNF function chain of the slice i is represented as $N^{Vi} = \left\{f_1^i, f_2^i, \cdots, f_K^i\right\}$.

In the computation resource pool on the access network side, each BBU provides the computation resource for RRU to process baseband data. The BBUs in the computation resource pool can share their own resources, and all computation resources are centralized to form a virtual computation resource pool through virtualization operation. In the access network, a time-varying random channel model is considered. This model assumes that there are multiple RRUs in a specific area. The total bandwidth of W Hz is divided into multiple physical resource blocks PRBs. The bandwidth of each PRB is w. These PRBs is shared by all RRUs. The entire network provides I slice service for U users in total, and the set of users is $\mathbf{U} = \{1, \cdots, U\}$. Each RRU can provide services for multiple types of slices. H is the set of limited channel states, $h_u^i(t) \in H = \{h_1, h_2, \cdots, h_H\}$ and $\sum_{m=1}^{H} P(h_m) = 1$, $h_{u,i}(t)$ is the channel gain when user u accesses slice i at time slot t, where $P(h_m)$ represents the probability that the channel state is h_m. When a user requests to access each slice, it is assumed that the channel state in each time slot is fixed, but the channel state between different time slots changes randomly, and the channel state among different time slots is independent of each other. $\overline{h^i}(t)$ represents the average channel gain when the user accesses slice i, and $\varepsilon^i(t)$ represents the spectral efficiency corresponding to time slot t.

In this paper, we construct the corresponding queue for data packets of each network slice service. We consider a discrete-time queuing system on the access network side. The length of each time slot is fixed, and multiple slices can be accessed in any time slot. $X_u^i(t)$ represents the number of data packets arriving in time slot t of the network slice i accessed by the user u, and the number of arriving data packets follows Gaussian distribution $E\left\{X_u^i(t)\right\} = \lambda_u^i$ and is independently and identically distributed among different time slots. The queue length of slice i at the beginning of time slot t is $Q^i(t)$, and $Q^i(t) = \sum_{u \in U} Q_u^i(t)$, where $Q_u^i(t)$ is the queue length of the slice i of the user u at time slot t.The dynamic update process of $Q^i(t)$ can be expressed as:

$$Q^i(t+1) = \max\left[Q^i(t) - D^i(t), 0\right] + X^i(t) \tag{1}$$

The number of data packets leaving the queue of slice i in time slot t is expressed as $D^i(t) = \varepsilon^i(t) \cdot w \cdot A^i(t)/S$, where $A^i(t)$ is the number of PRBs allocated to slice i by the network in time slot t, and S is the size of data packets in the slice queue. $X^i(t) = \sum_{u \in U} X_u^i(t)$ is the number of data packets arriving at slice i in time slot t. Let $Q(t) = \left\{Q^1(t), Q^2(t), \cdots, Q^I(t)\right\}$ represent the global queue status information of the system in time slot t, and $H(t) = \left\{\overline{h^1}(t), \overline{h^2}(t), \cdots, \overline{h^I}(t)\right\}$ represent the global channel status information in time slot t.

On the access network side, the set of computation resource is composed of multiple CPU cores, and the total number of CPU cores in the computation resource pool is Y. Each CPU core has the same data bandwidth processing capability, which is b Mbps. Let $\sigma^i(t)$ represent the computation resource allocation strategy in time slot t, and $\sigma^i(t)$ satisfies

$$\sigma^i(t) \geq 0, \sum_{i \in \mathbf{I}} \sigma^i(t) \leq Y \tag{2}$$

Similarly, let $\omega^i(t)$ represent the PRB allocation strategy in time slot t, $\omega^i(t)$ satisfies

$$\omega^i(t) \geq 0, \sum_{i \in \mathbf{I}} \omega^i(t) \leq Z \tag{3}$$

where Z is the total number of PRBs in the network.

The sum rate of all network slices of the entire network in time slot t can be expressed as

$$r(t) = \sum_{i \in \mathbf{I}} r^i(t) = \sum_{i \in \mathbf{I}} \varepsilon^i(t) \cdot w \cdot \omega^i(t) \tag{4}$$

On the core network side, define $\lambda_n(t) \in \{0,1\}$ to indicate the working status of the node, let $\lambda_n = 1$ indicate that node n is operating normally, otherwise the node fails. Define $l_{n,n'}(t) \in \{0,1\}$ to indicate the working status of the link between nodes n and n', let $l_{n,n'} = 1$ indicate that the link between node n and n' is operating normally, otherwise the link fails.

Define a linear relationship between the number of computation resources C_k^i required by VNF f_k^i and the required data processing rate R_k^i [12]. Let α_k be the correlation coefficient between the VNF computation resource requirement and the data processing rate, then the computation resource requirement of VNF f_k^i is expressed as

$$C_k^i = \alpha_k R_k^i \tag{5}$$

A binary node association factor $\beta_{k,n}^i \in \{0,1\}$ is defined to represent the mapping relationship between VNFs and physical nodes, if and only if the VNF f_k^i is needed in the VNR i, and the VNF f_k^i is mapped to the node n, $\beta_{k,n}^i = 1$. Assuming that there is a queuing process for the arriving data flow of each VNF mapped to the physical node, let $Q_{k,n}^i(t)$ represent the queue length of VNF f_k^i mapped to node n at the beginning of time slot t, and represent the number of data packets arriving at VNF f_k^i of slice i in time slot t. Similar to the access network side, the number of arriving data packets also follows Gaussian distribution $E\left\{X_u^i(t)\right\} = \lambda_u^i$ and is independently and identically distributed among different time slots. R_k^i is the data processing rate required by VNF f_k^i. The dynamic update process of VNF VNF f_k^i queue of node n can be expressed as

$$Q_{n,k}^i(t+1) = \max\left[Q_{n,k}^i(t) - R_k^i(t), 0\right] + X_k^i(t) \tag{6}$$

The data packet of VNF f_k^i which leave the f_k^i queue mapped to the node n is expressed as

$$L_{k,n}^i = \min\left[R_k^i, Q_{k,n}^i\right] \tag{7}$$

Let $B_{n,n'}$ denote the bandwidth consumption between physical nodes n and n', which can be expressed as

$$B_{n,n'} = \sum_{i \in \mathbf{I}} \sum_{f_k^i, f_j^i \in N^{Vi}} \beta_{j,n}^i \beta_{k,n'}^i S \cdot L_{j,n}^i(f_j | f_k, i) \tag{8}$$

3 Problem Formulation

The optimization goal of this model is to maximize the overall average network utility on a long-term scale of end-to-end network slicing, maximize the average network revenue and minimize the average deployment cost while ensuring system transmission rate. The revenue comes from the service rate, and the expenditure comes from the VNF deployment cost, which can be expressed as

$$g_r^i = \overline{r^i}\delta_r \tag{9}$$

$$l_{CPU}^i = \sum_{f_k^i \in N^{Vi}} \sum_{n \in N^P} \beta_{k,n}^i \overline{C}_k^i \delta_n \tag{10}$$

$$l_B^i = \overline{B}_{n,n'}^i \delta_B \tag{11}$$

where δ_r, δ_n, δ_B respectively represent the unit price of the service rate, the unit price of the computation resource on the server node, and the unit price of the communication resource between the nodes.

The overall average network utility can be expressed as

$$\bar{u} = g_r - l_{CPU} - l_B \tag{12}$$

Therefore, the optimization goal of this paper can be expressed as

$$\max_{\sigma(t),\omega(t),\Psi(t),\beta(t)} \bar{u} \tag{13a}$$

$$\text{s.t. } \sigma^i(t) \geq 0, \sum_{i \in I} \sigma^i(t) \leq Y, \forall t \tag{13b}$$

$$\omega^i(t) \geq 0, \sum_{i \in I} \omega^i(t) \leq Z, \forall t \tag{13c}$$

$$\omega^i(t)\, r^i(t) \geq R_{rsv}^i, \forall i \in \mathbf{I}, \forall t \tag{13d}$$

$$\bar{C}_n^{\pi}(c) \leq C_n^{\max}, \forall n \in N^P \tag{13e}$$

$$\bar{B}_{n,n'}^{\pi}(c) \leq B_{n,n'}, \forall n, n' \in N^P \tag{13f}$$

$$\sum_{n \in N^P} \beta_{k,n}^i = \rho_k^i, \forall i \in \mathbf{I}, f_k^i \in N^{Vi} \tag{13g}$$

Constraint (13b) indicates that the allocation of computation resources shall not exceed the total computation resources. Constraint (13c) indicates that the total link bandwidth occupied by the data streams of each slice in the access network during transmission should not exceed the upper limit of the access network bandwidth. Constraint (13d) ensures that each slice is guaranteed a minimum data rate. Constraint (13e) indicates that the total computation resources required by the VNF mapped to the same node n cannot exceed the total computation resources of the node n; Constraint (13f) means that when VNF f_k^i is mapped to node n and VNF f_{k+1}^i is mapped to node n', the mapping from virtual link to physical link is realized, and the link bandwidth required by any virtual network function cannot exceed the maximum available bandwidth provided by any two nodes, where $B_{n,n'}$ is the upper limit of the maximum available bandwidth provided between any two physical nodes. Constraint (13g) indicates that for each slicing requirement, the VNF required by the slice should be mapped to the physical node, where ρ_k^i indicates whether slice i requires VNF f_k^i, if necessary, $\rho_k^i = 1$; otherwise, $\rho_k^i = 0$.

4 Joint Allocation of Two-Stage Communication and Computation Resource Based on DQN

In this paper, a two-stage DQN algorithm is used to jointly allocate dynamic communication and computation resource in end-to-end network slices. The optimization problem to be solved can be modeled as a constrained Markov decision process(CMDP) problem, which can be described by a four-tuple $\langle C, A, p_a\left(c'|c\right), \ R_a\left(c'|c\right)\rangle$, where C is a finite set of states of the system, and A is a finite set of actions that can be taken. $p_a\left(c'|c\right)$ is the probability of state c transitioning to c' after performing action a in state in current time slot t. $R_a\left(c'|c\right)$ is the reward function of state transfer to c' after system executes action a in state c, representing the instant cost/reward, which represents the learning goal.

4.1 DQN Training Algorithm

DQN algorithm takes state c as input and outputs the corresponding action after neural network analysis. The essence of the algorithm is to approximate the distribution of Q value by using neural network training function f_{ap}. Q value can be expressed as

$$\mathbf{Q}\left(c,a\right) \approx f_{ap}\left(c,a,\theta\right) \tag{14}$$

where θ represents the weight of the main network, $\mathbf{Q}\left(c,a\right) = \left[Q\left(c,a_1\right), Q\left(c,a_2\right), \ \cdots, Q\left(c,a_T\right)\right]$. The main network model used in this paper is CNN, and the network structure mainly includes an input layer, a convolutional layer, and a fully connected layer.

The target Q value y is expressed as

$$y = r\left(c,a\right) + \gamma\left[\max_{a'\in A} Q\left(c',a',\theta^-\right)\right] \tag{15}$$

where θ^- represents the weight of the target Q network. To fit complex environmental characteristics, it is necessary to repeatedly learn and train the weight function to improve the performance of network prediction. The training process of DQN algorithm is shown in Fig. 2. In this training model, the weight θ is optimized by minimizing the loss function between the main network and the target Q network. The loss function can be expressed as

$$L\left(\theta\right) = E\left[\left(y - Q\left(c,a,\theta\right)\right)^2\right] \tag{16}$$

After the main network of DQN algorithm undergoes the training process, the optimal allocation scheme of network slice resources can be obtained by using the trained main network. The global process of the two-stage learning algorithm proposed in this paper is: randomly generate a CPU resource and PRB allocation method $\sigma(t)$ and $\omega(t)$ that satisfy the constraints (13b)–(13d). Based on $\sigma(t)$ and $\omega(t)$, execute one-stage algorithm to get VNF migration decision $\Psi(t)$ and VNF resource allocation strategy $\beta(t)$ and update; based on $\Psi(t)$ and $\beta(t)$, the new CPU resource and PRB allocation method $\sigma(t)$ and $\omega(t)$ are obtained by executing the two-stage algorith. The final PRB and CPU resource allocation method, VNF migration decision and VNF resource allocation strategy are obtained.

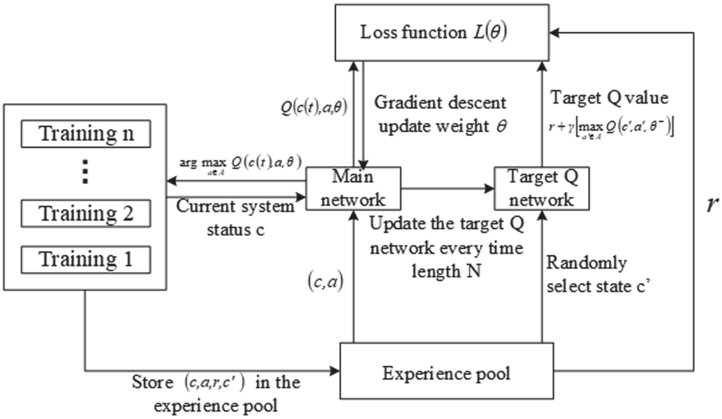

Fig. 2. DQN training model.

4.2 CN-Side Algorithm

The one-stage algorithm is a DQN-based CN-side communication and computation resource joint allocation algorithm. In the core network, the dynamic allocation of CPU resource on servers in the physical network, the dynamic allocation of link bandwidth resource between servers, and the migration strategy of VNF are described as a CMDP problem. A basic element is defined, which includes four tuples of system state, migration behavior, state transition probability and cost function. The state of the system in time slot t is defined as$c_t = \{Q(t), \lambda(t), l(t)\} \in C$, and the action in time slot t is defined as $a_t = \{\Psi(t), \beta(t)\} \in A$, where $\Psi(t)$ is the two-dimensional migration action vector of the VNF in time slot t. The decision is composed of $\phi_{f_k^i}(t)$ and $n_{f_k^i}(t)$, $\forall f_k^i \in N^{Vi}$, $i \in I$, $\phi_{f_k^i}(t) = 1$ means that VNF f_k^i of slice i is migrated in time slot t, otherwise $\phi_{f_k^i}(t) = 0$; $n_{f_k^i}(t)$ is the target node of VNF f_k^i migration; $\beta(t)$ is the set of mapping actions for each VNF in time slot t.

The process of mapping state space to action space is defined as $\pi : \mathrm{C} \rightarrow A$, which is a stability strategy, that is, $a = \pi(c)$. According to strategy $\pi \in \Pi$, the expected cumulative CPU resource allocation in time slot t is as follows

$$\begin{aligned} \bar{C}^{\pi}(c) &= E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t C(c_t, a_t) | c_0 = c\right\} \\ &= E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t \sum_{f_k^i \in N^{Vi}} \sum_{n \in N^P} \beta_{k,n}^i(t) \alpha_k R_k^i | c_0 = c\right\} \end{aligned} \tag{17}$$

The expected cumulative bandwidth resource allocation is as follows

$$\begin{aligned} \bar{B}_{n,n'}^{\pi}(c) &= E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t B_{n,n'}(c_t, a_t) | c_0 = c\right\} \\ &= E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t \sum_{i \in I} \sum_{f_k^i, f_j^i \in N^{Vi}} \beta_{j,n}^i \beta_{k,n'}^i S \cdot L_{j,n}^i(f_k | f_j, i) | c_0 = c\right\} \end{aligned} \tag{18}$$

where $\gamma \in [0, 1)$ is a discount factor, indicating the degree of attenuation of the reward function value, which indicates the degree of influence of future rewards on the current behavior choice. The core network side optimization goal is to find the optimal VNF migration decision $\Psi(t)$ and VNF resource allocation strategy $\beta(t)$ to minimize deployment costs. The stochastic optimization model can be expressed as

$$\min_{\Psi(t), \beta(t)} \bar{C}^{\pi}(c) + \sum_{n \in N^P} \bar{B}_{n,n'}^{\pi}(c) \tag{19}$$

Combined with the definition and description of the Markov decision problem above, the state, action, and reward of the one-stage algorithm are defined as follows:

State: $c_t = \{Q(t), \lambda(t), l(t)\} \in C$.

Action: A set of VNF migration and resource allocation actions that satisfy $[\Psi^*(t), \beta^*(t)] = \arg \min_{\Psi(t), \beta(t)} \bar{C}^{\pi}(c) + \sum_{n \in N^P} \bar{B}_{n,n'}^{\pi}(c)$.

Reward: We define the reward function as the respective utility of each slice. If the action selection does not satisfy the constraints (13b)-(13g), set the reward function to fixed –1.

$$r^i(c, a) = \begin{cases} u^i, C13a - C13f \\ -1, otherwise \end{cases} \tag{20}$$

The one-stage algorithm is shown in Algorithm 1.

4.3 RAN-Side Algorithm

The two-stage algorithm is a DQN-based RAN-side communication and computation resource joint allocation algorithm. In the access network, the state of the system in time slot t is defined as $c_t = (Q(t), H(t)) \in C$, and the action in time slot t is defined as $a_t = (\sigma(t), \omega(t)) \in A$.The process of mapping state space to action space is defined as $\pi : \mathrm{C} \rightarrow A$, which is a stability strategy, that

Algorithm 1. CN-side algorithm

Input:

Physical network topology G^P, Virtual network topology G^V, PRB and CPU resource allocation method $\sigma(t), \omega(t)$;

1: for t=1, 2, ..., T do
2: Monitor the global status c_t of the core network side in current time slot t, including global queue status information $Q(t)$, global node status $\lambda(t)$, and global link status $l(t)$
3: if $\lambda_n(t) = 0$ or $l_{n,n'}(t) = 0$
4: Calculate the optimal VNF migration strategy $\Psi(t)$ and VNF communication and computation resource allocation strategy $a_t^* = \arg\min_{a \in A} Q(c_t, a, \theta)$ on the basis of migrating all $\forall f_k^i \in N^{Vi}$ satisfying $\beta_{k,n}^i = 1$ to other nodes
5: else
6: Directly calculate the optimal VNF migration strategy $\Psi(t)$ and VNF communication and computation resource allocation strategy $a_t^* = \arg\min_{a \in A} Q(c_t, a, \theta)$
7: Perform VNF migration based on the optimal action a_t^*, and allocate communication and computation resource
8: t=t+1
9: end for

Output:

VNF migration strategy $\Psi(t)$ and VNF resource allocation strategy $\beta(t)$

is, $a = \pi(c)$. According to strategy $\pi \in \Pi$, the expected cumulative slice sum rate is

$$\begin{aligned}\bar{r}^{\pi}(c) &= E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t r(c_t, a_t) | c_0 = c\right\} \\ &= E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t \sum_{i \in I} \varepsilon^i(t) \cdot w \cdot \omega^i(t) | c_0 = c\right\}\end{aligned} \tag{21}$$

The objective of access network side optimization is to find the optimal allocation scheme of PRB and computation resources under the premise of satisfying the constraints of the minimum service rate of each network slice and the constraints of network bandwidth resource, so as to maximize the network revenue. The stochastic optimization model can be expressed as

$$\max_{\sigma(t), \omega(t)} E^{\pi}\left\{\sum_{t=0}^{\infty} \gamma_t \sum_{i \in I} \varepsilon^i(t) \cdot w \cdot \omega^i(t) | c_0 = c\right\} \tag{22}$$

The state, action, and reward of the two-stage algorithm are defined as follows:

State: $c_t = (Q(t), H(t)) \in C$.

Action: A set of PRB and computation resource allocation actions that satisfy $[\sigma^*(t), \omega^*(t)] = \arg\max_{\sigma(t), \omega(t)} \bar{r}^{\pi}(c)$.

Reward: When the constraints (4-13b)-(4-13g) are satisfied, the reward function is defined as the sum of system utility obtained after the slices select their respective PRBs and computation resource, otherwise, it is defined as a negative feedback.

$$r\left(c,a\right)=\begin{cases}\sum\limits_{i\in I}u^{i},C13a-C13f\\ \quad -1,otherwise\end{cases} \tag{23}$$

The two-stage algorithm is shown in Algorithm 2. The core network-side VNF migration strategy and mapping method obtained by the one-stage algorithm are taken as the input of the two-stage algorithm to obtain the access network-side CPU resources and PRB allocation result. The entire algorithm will reach the end of the slice life cycle to obtain the final VNF migration strategy, VNF resource allocation strategy, PRB allocation strategy and CPU resource allocation method. The resource allocation process ends.

Algorithm 2. RAN-side algorithm

Input:

VNF migration strategy $\Psi\left(t\right)$ and VNF resource allocation strategy $\beta\left(t\right)$;

1: for t=1, 2, ..., T do
2: Monitor the global status c_t of the access network side in the current time slot, including global queue status information $Q\left(t\right)$ and global channel status information $H\left(t\right)$
3: Calculate the optimal PRB and computation resource allocation actions $a_t^* = \arg\max\limits_{a\in A} Q\left(c_t,a,\theta\right)$
4: Adjusting the PRB and computation resource allocation of radio access network slices based on the optimal action a_t^*
5: t=t+1
6: end for

Output:

PRB allocation strategy $\omega\left(t\right)$ and CPU resource allocation method $\sigma\left(t\right)$

5 Performance Evaluation

In the simulation, this paper assumes that the network scenario is a fully connected network, and the infrastructure has $N = 10$ general-purpose processors. Considering 3 types of network slices with different minimum rate requirements, 8 types of VNFs constitute the network function chain of the slices and the computation resource demand coefficient $\alpha_m = 0.1$. The service function chain of each slice includes multiple VNFs, and the number of VNFs follows the uniform distribution of 6 to 8. The arrival process of sliced data packets is independent and identically distributed Poisson distribution, and the node failure rate and link failure rate of the underlying network are uniformly distributed. The remaining simulation parameters are shown in Table 1.

The main network and the target Q network in the two-stage resource allocation algorithm based on DQN used in this paper are multi-layer convolutional CNN networks, including 3 layers of convolutional layers and 2 layers of fully connected layers. The parameters of the target Q network are updated every 200 iterations. Let the capacity of DQN experience playback pool in the training process be 10000, and the probability value of $\varepsilon-greedy$ strategy $\varepsilon = 0.7$.

Table 1. Simulation Parameters

Parameter	Value
Number of RRU CPU core Y	30
Number of PRB Z	50
System bandwidth W	10 MHz
Pathloss from RRU to user	$37.6 \log 10 (d\, [km]) + 128.1$ dB
Noise power spectral density	-174 dBm/Hz
Maximum RRU transmit power $P_{\max}$	39 dBm
Number of server nodes N	10
Number of VNF types	8
Minimum rate of slice i R^i_{rsv}	10 Mbps, 5 Mbps, 51 Mbps
Maximum queue length of each slice	20 packets
Data packet size S	4 kbit/packet
Time slot length	1 ms
Discount factor γ	0.9
Maximum number of iterations	2000
Learning rate	0.0001

To evaluate the feasibility of the model and the effectiveness of the algorithm, the overall network utility of the network slicing service, the average slicing rate, and the total cost of the network slicing system are used as performance evaluation indicators, and the performance of the algorithm proposed in this paper is compared with that of the proportional fair static sharing(PFSS) algorithm and the heuristic algorithm.

When the number of users is 30 and the unit price of the service rate is 7, the comparison results of network utility on the long-term scale obtained by different resource allocation algorithms are shown in Fig. 3. As the time series progresses, network utility tends to stabilize. Compared with the PFSS algorithm and heuristic algorithm, the algorithm proposed in this paper can obtain the maximum network utility.

Figure 4 shows the change in the average total slice rate of the three comparison algorithms as the number of users increases when the unit price of the service rate is 7. The simulation result shows that the average total slice rate corresponding to the three algorithms all increase with the increase of the number of users. When the number of users is less than 10, the heuristic algorithm can maintain the same total rate as the algorithm proposed in this paper; but as the number of users continues to increase, the rate of the heuristic algorithm gradually increases due to the limitation of the general service node resources. The two-stage dynamic resource allocation algorithm based on DQN proposed in this paper can guarantee the best system service rate.

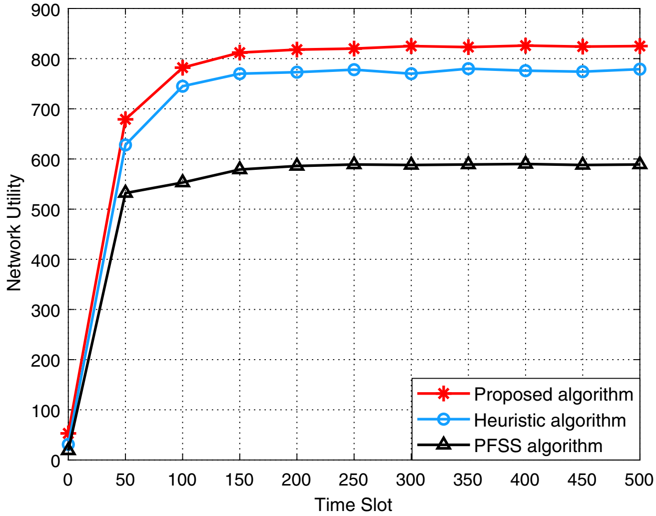

Fig. 3. Comparison of network utility of three algorithms.

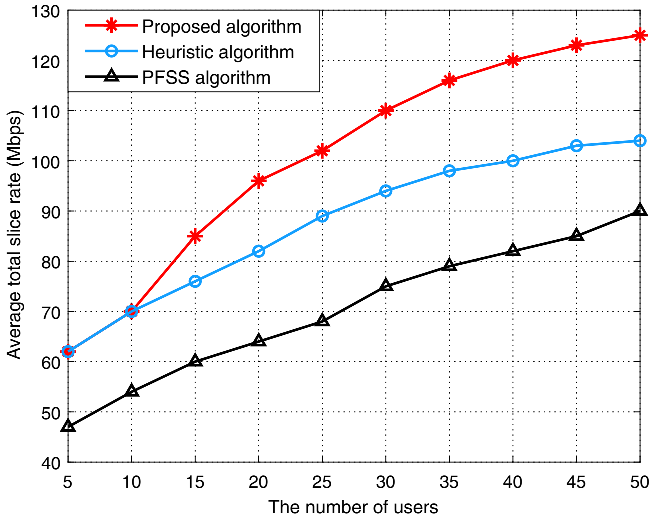

Fig. 4. Average total slice rate versus the number of users.

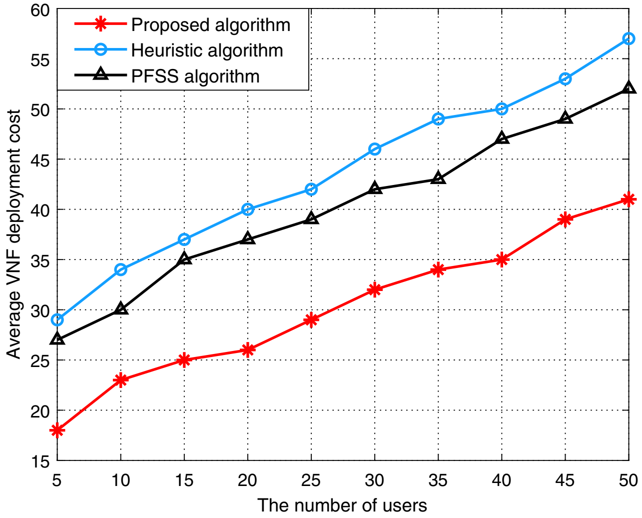

Fig. 5. Average VNF deployment cost versus the number of users.

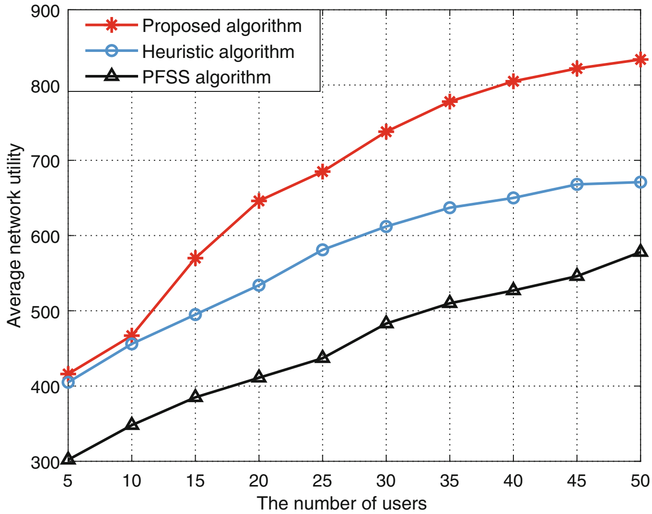

Fig. 6. Average network utility versus the number of users.

Figure 5 illustrates the change of the average slice VNF deployment cost with the increase in the number of users in the three comparison algorithms when the unit price of the service rate is 7. The simulation results show that the VNF deployment costs corresponding to the three algorithms all increase with the increase in the number of users. As can be seen from the figure, the algorithm proposed in this paper can achieve the lowest total system cost.

Figure 6 shows the change in the average network utility as the number of users increases when the service rate unit price is 7, and the number of users in each slice is as uniform as possible. The simulation results show that when the number of users increases, the average network utility increases. Compared with the PFSS algorithm and heuristic algorithm, the algorithm proposed in this paper can obtain the largest average network utility.

6 Conclusions

This paper proposes a joint allocation method of dynamic communication and computation resource in an end-to-end network slicing scenario, aiming at the dynamic changes of network slicing data queues, radio channel status, and physical network topology, constructs the end-to-end slicing dynamic resource allocation model. With the optimization goal of maximizing the overall utility of the network on a long-term scale, the dynamic joint allocation of communication and computation resources is performed on the network slicing. According to the service requirements of network slicing, the flexible allocation of communication and computation resources is realized in discrete time slots according to the current system state. This paper proposes a two-stage communication and computation resource joint allocation algorithm based on DQN. The simulation results show that the scheme proposed in this paper can achieve the purpose of optimizing the overall utility of the network, increasing the long-term average revenue, and reducing the average cost of the system.

References

1. Chien, H.T., Lin, Y.D., Lai, C.L., et al.: End-to-end slicing with optimized communication and computing resource allocation in multi-tenant 5G systems. IEEE Trans. Veh. Technol. **69**(2), 2079–2091 (2020)
2. Ordonez-Lucena, J., Ameigeiras, P., Lopez, D., et al.: Network slicing for 5G with SDN/NFV: concepts, architectures and challenges. IEEE Commun. Mag. **55**(5), 80–87 (2017)
3. Nakao, A., Du, P., Kiriha, Y., et al.: End-to-end network slicing for 5G mobile networks. J. Inf. Process. **25**, 153–163 (2017)
4. Riggio, R., Bradai, A., Harutyunyan, D., et al.: Scheduling wireless virtual networks functions. IEEE Trans. Netw. Serv. Manage. **13**(2), 240–252 (2016)
5. Tong, Z., Zhang, T., Zhu, Y., et al.: Communication and computation resource allocation for end-to-end slicing in mobile networks. In: IEEE/CIC International Conference on Communications in China (ICCC) 2020, China, pp. 1286–1291 (2020). https://doi.org/10.1109/ICCC49849.2020.9238794
6. Zhou, L., Zhang, T., Li, J., et al.: Radio resource allocation for RAN slicing in mobile networks. In: IEEE/CIC International Conference on Communications in China (ICCC) 2020, China, pp. 1280–1285 (2020). https://doi.org/10.1109/ICCC49849.2020.9238905
7. Qiang, Y., Zhuang, W., Shan, Z., et al.: Dynamic radio resource slicing for a two-tier heterogeneous wireless network. IEEE Trans. Veh. Technol. 1 (2018)

8. Florian, B., Ramos-Cantor, O.D., Steffen, H., et al.: Optimized cell planning for network slicing in heterogeneous wireless communication networks. IEEE Commun. Lett. **22**(8), 1676–1679 (2018)
9. Jin, H., Lu, H., Jin, Y., et al.: IVCN: information-centric network slicing optimization based on NFV in fog-enabled RAN. IEEE Access **7**, 69667–69686 (2019)
10. Xu, Q., Wang, J., Wu, K.: Learning-based dynamic resource provisioning for network slicing with ensured end-to-end performance bound. IEEE Trans. Netw. Sci. Eng. 28–41 (2018)
11. Zhang, J., Wu, M., Ma, R., et al.: Dynamic resource sharing scheme across network slicing for multi-tenant C-RANs. In: IEEE/CIC International Conference on Communications in China (ICCC Workshops) 2018, China, pp. 172–177 (2016). https://doi.org/10.1109/ICCChinaW.2018.8674479
12. Jian, Y., Zhang, S., Wu, X., et al.: Online learning-based server provisioning for electricity cost reduction in data center. IEEE Trans. Control Syst. Technol. **25**(3), 1044–1051 (2017)